

# **DISEÑO DE REDES NEURONALES ARTIFICIALES NO SUPERVISADAS ORIENTADAS A LA INTELIGENCIA DE NEGOCIO**

**Alumno: Germán Álvarez Alba**  
**Tutor: Diego Andina de la Fuente**

## **TRIBUNAL**

**Presidente: Seijas Martínez-Echevarría, Juan Isidoro**  
**Vocal: Alarcón Mondéjar, Martín Javier**  
**Secretaría: Torres Alegre, Santiago**  
**Suplente: Córdoba Herralde, Ricardo de**

**Grado en Ingeniería de Tecnologías y Servicios de Telecomunicación**

**DEPARTAMENTO DE SEÑALES, SISTEMAS Y RADIOCOMUNICACIONES**

**UNIVERSIDAD POLITÉCNICA DE MADRID**



**TRABAJO DE FIN DE GRADO**

**Fecha:**

**Calificación:**

**DISEÑO DE REDES NEURONALES ARTIFICIALES NO  
SUPERVISADAS ORIENTADAS A LA INTELIGENCIA DE  
NEGOCIO**

**Alumno: Germán Álvarez Alba  
Tutor: Diego Andina de la Fuente**

**TRIBUNAL**

**Presidente: Seijas Martínez-Echevarría, Juan Isidoro  
Vocal: Alarcón Mondéjar, Martín Javier  
Secretaría: Torres Alegre, Santiago  
Suplente: Córdoba Herralde, Ricardo de**

**Grado en Ingeniería de Tecnologías y Servicios de Telecomunicación**

**DEPARTAMENTO DE SEÑALES, SISTEMAS Y RADIOCOMUNICACIONES**

## RESUMEN

El cerebro humano es el sistema de cálculo más complejo que conoce el hombre, de hecho existe el mito de que no usamos ni el 10% de su capacidad. Cada vez son más frecuentes los programas computacionales que se inspiran en el funcionamiento del cerebro para realizar automáticamente tareas que percibimos como inteligentes.

Estos programas están inspirados en la capacidad del cerebro para pensar, recordar, reconocer y resolver problemas, y poder obtener información de una serie de datos para nuestro interés.

Existen varios tipos de redes neuronales artificiales, que se pueden clasificar en dos grandes grupos: de aprendizaje supervisado y no supervisado. Entre las pertenecientes al segundo, se pueden destacar las redes de Teuvo Kohonen y la *Koniocortex Like Network*, las pruebas experimentales se han realizado sobre la primera red, introduciendo finalmente el uso de la segunda.

La red de Kohonen está basada en mapas autorganizados descubiertos a nivel cerebral. Este tipo de red posee un aprendizaje no supervisado competitivo, es decir, que no existe un supervisor externo que dé como correcta o incorrecta la operación que haya realizado la red porque no se dispone de ninguna salida hacia la cual la red neuronal debe tender. El procedimiento que seguirá la red es, de manera auto-organizada, agrupar o mapear rasgos comunes, regularidades, categorías en los datos de entrada, etc., e incorporarlos a su estructura interna de conexiones.

En este trabajo, a efectos ilustrativos, se ha comenzado a utilizar esta red para clasificar varias plantas de Iris, donde se le pasan 150 patrones y la red los organiza activando a tres neuronas diferentes.

Una vez entendido el funcionamiento de la red de Kohonen, aplicaremos esta red para organizar una serie de datos bancarios y obtener información que pudiera ser aplicada para obtener mayor rendimiento en el negocio de la banca.

**Palabras clave:** Mapas, neuronas, aprendizaje, categorías, Kohonen.

## ABSTRACT

The human brain is the most complex calculation system known to man, in fact there is a myth that we do not even use 10% of its capacity. Nowadays it's becoming more frequent computer programs inspired by brain functions to automatically perform tasks that we perceive as intelligent.

These programs are inspired by the brain's ability to think, remember, recognize, solve problems and to obtain information from a number of data for our interest.

There are several types of artificial neural networks, which can be classified into two groups: supervised and unsupervised learning. Among those belonging to the second, you can highlight Teuvo Kohonen networks and Koniocortex Like Network, experimental tests have been conducted on the first network, finally introducing the use of the second.

Kohonen network is based on self-organizing maps discovered on the brain. This type of network has an unsupervised competitive learning, this means that there is no external supervisor for correct operations made by the network because it does not have any output to which the neural network must tend. The procedure followed by the network is self-organized, grouping or mapping common features, regularities, categories in the input data, etc., and incorporate them into its internal structure of connections.

In this paper, for illustrative purposes, it has begun to use this network to classify various Iris type plants, in total there were 150 patterns and the network had to organize those patterns into three categories.

Once understood the operation of the Kohonen network, this network will be applied to organize a series of bank data and information that could be applied for better performance in the banking business.

**Keywords:** Maps, neurons, learning, categories, Kohonen.

## Índice general

<b>1. Computación neuronal</b>	7
1.1 Introducción	7
1.2 Características de las redes neuronales artificiales	7
1.3 Estructura básica de una red neuronal	8
1.3.1 Analogía con el cerebro humano	8
1.3.2 Redes neuronales artificiales (ANN)	9
1.4 Programación neuronal	10
1.5 Historia de la computación neuronal	11
1.6 Aplicaciones de las redes neuronales artificiales	14
<b>2. Fundamentos de las redes neuronales artificiales</b>	16
2.1 Redes neuronales artificiales	16
2.2 Tipos de redes neuronales	19
<b>3. Red Self Organizing Map (SOM)</b>	20
3.1 Ideas intuitivas sobre el algoritmo del SOM	21
3.2 Fundamentos biológicos	21
3.3 Arquitectura del SOM	22
3.4 Algoritmo del SOM	23
<b>4. Business Intelligence</b>	24
4.1 Introducción	24
4.2 ¿Qué es <i>Business Intelligence</i> ?	24
4.3 Beneficios que aporta el BI	26
4.4 Componentes de Business Intelligence	27
4.4.1 Fuentes de información	28
4.4.2 Proceso de extracción, transformación y carga (ETL)	29
4.4.3 <i>Datawarehouse</i> o almacén de datos	30
4.4.4 Herramientas y técnicas	30
<b>5. Aplicación de la red de Kohonen a <i>Credit Scoring</i></b>	33
5.1 ¿Qué es <i>Credit Scoring</i> ?	33
5.2 Ejemplo de aplicación	34
<b>6. Discusión</b>	41
<b>7. Conclusión</b>	44
<b>8. Referencias</b>	45

## Índice de figuras

Figura 1: Estructura de una neurona.....	9
Figura 2: Estructura básica de una red neuronal.....	10
Figura 3: Perceptrón de Rosenblatt .....	12
Figura 4: Modelo “ <i>Brain State-in-a-Box</i> ” .....	13
Figura 5: Esquema básico de funcionamiento de una neurona.....	17
Figura 6: Esquema de la arquitectura de una red .....	18
Figura 7: Principales ANN a lo largo de la historia .....	19
Figura 8: Estructura modelo SOM .....	22
Figura 9: Componentes principales en un proceso de BI .....	25
Figura 10: Esquema BI .....	27
Figura 11: Fuentes de información principales en un proceso BI.....	29
Figura 12: Datos bancarios no normalizados.....	34
Figura 13: Esquema de la red neuronal compuesta por 14 entradas y 8 salidas...36	
Figura 14: Salidas de la red neuronal para el caso de Australia .....	36
Figura 15: Datos de entrada para el caso de Alemania.....	39
Figura 16: Esquema de la red neuronal compuesta por 24 entradas y 6 salidas...40	
Figura 17: Salidas de la red neuronal para el caso de Alemania.....	40
Figura 18: Curva de la plasticidad .....	42
Figura 19: Curva de activación de las neuronas .....	42
Figura 20: Red KLN .....	43

## 1. Computación neuronal

### 1.1 Introducción

El cerebro humano es el sistema de cálculo más complejo que conoce el hombre. El ordenador y el hombre realizan clases de tareas diferentes en sencillez dependiendo del que lo ejecute. Por ejemplo, un ordenador es capaz de llevar los libros de cuentas de una empresa con una asombrosa facilidad mientras que para el cerebro es una tarea difícil, por otro lado, reconocer el rostro de una persona el cerebro lo hace rápidamente, sin embargo para el ordenador resulta ser una tarea difícil.

Esta capacidad del cerebro para pensar, recordar, reconocer y resolver patrones ha inspirado a muchos científicos a intentar modelar en un ordenador el funcionamiento del cerebro humano.

Un grupo de investigadores ha realizado la creación de un modelo en el ordenador que adopte o iguale algunas de las funciones básicas del cerebro. El resultado ha sido una nueva tecnología llamada Redes Neuronales Artificiales.

Este gran interés que está habiendo estos últimos años en este campo es debido a la gran cantidad de aplicaciones que puede tener, por ejemplo, y más centrado en este trabajo, la aplicación de las redes neuronales a reconocer patrones y ser capaz de clasificarlos siguiendo una serie de características comunes.

### 1.2 Características de las redes neuronales artificiales

Las redes neuronales artificiales, ANN, (*Artificial Neural Networks*) están inspiradas en las redes neuronales biológicas del cerebro humano. Estas, están formadas por elementos que se comportan de forma parecida a la neurona biológica en sus funciones más básicas lógicamente. La organización es similar a la que presenta el cerebro humano.

Las ANN al margen de “parecerse” al cerebro presentan una serie de características propias del cerebro. Por ejemplo, las ANN aprenden de la experiencia, generalizan de ejemplos previos a ejemplos nuevos y abstraen las características principales de una serie de datos.

Se pueden distinguir tres funciones principales de estas redes ANN:

- **Aprender:** Consiste en adquirir conocimiento de cierta cosa a través de su estudio o ejercicio. Las ANN son capaces de cambiar de comportamiento en función del entorno. Simplemente, se les muestra un conjunto de entradas y ellas mismas se ajustan para producir una serie de salidas consistentes.

- **Generalizar:** Esta función consisten principalmente en extender o ampliar una cosa. Las ANN generalizan automáticamente debido a su propia estructura y naturaleza. Estas redes pueden ofrecer, dentro de un margen, respuestas correctas a entradas que presentan pequeñas variaciones debido a los efectos de ruido o distorsión.
- **Abstraer:** Se trata de aislar las cualidades de un objeto.

## 1.3 Estructura básica de una red neuronal

### 1.3.1 Analogía con el cerebro humano

La neurona, es la unidad fundamental del sistema nervioso y en particular del cerebro. Cada neurona es una simple unidad procesadora que recibe y combina señales desde y hacia otras neuronas. Si la combinación de entradas es suficientemente fuerte la salida de la neurona se activa.

Las partes principales de la neurona que se pueden observar en la figura son:

- **Cuerpo o soma:** Centro metabólico, lugar donde se fabrican las moléculas y se realizan las actividades fundamentales para mantener la vida y las funciones de la célula nerviosa.
- **Axón:** Prolongación del soma neuronal, generalmente más larga que las dendritas. Cada neurona tiene un único axón y a través de él se propaga la información a otras neuronas. La información sale del soma neuronal o las dendritas y se transmite a las neuronas con las que establece la sinapsis a través de los botones terminales situados en los extremos del axón.
- **Dendritas:** Son prolongaciones del soma neuronal con forma de árbol y son las principales áreas receptoras de información que llega a la neurona. La zona de transferencia de información de una neurona a otra es la sinapsis. La sinapsis tiene un componente presináptico y un componente postsináptico siendo la dendrita el postsináptico. El flujo de información va desde la zona presináptica a la zona postsináptica. La membrana de las dendritas tiene un elevado número de receptores que son moléculas especializadas sobre las que actúan los neurotransmisores desde otras neuronas.

Estas neuronas, densamente interconectadas entre sí forman el cerebro.



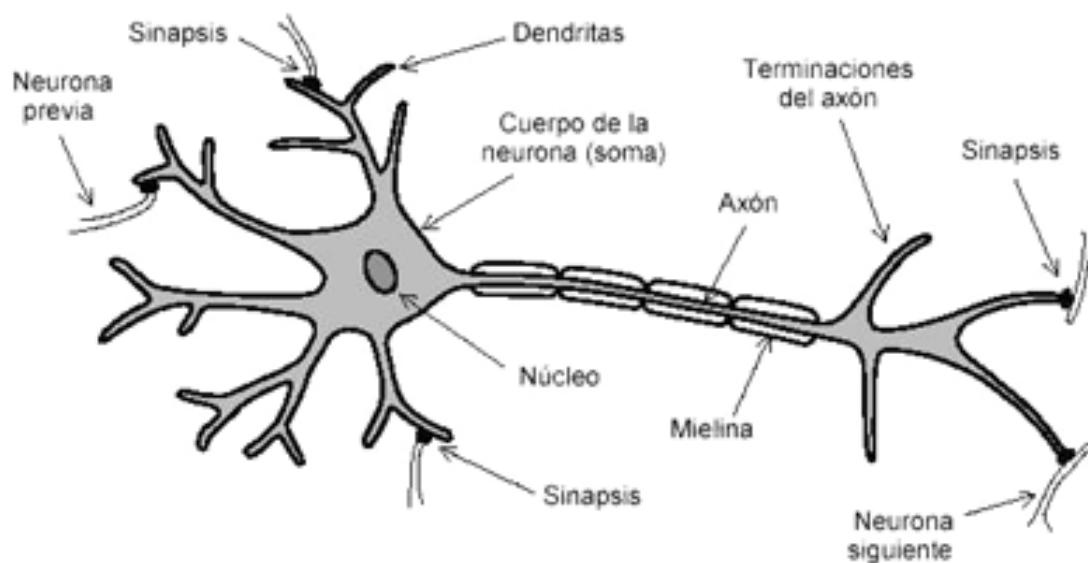


Figura 1. Estructura de una neurona

### 1.3.2 Redes neuronales artificiales (ANN)

En las ANN, la unidad análoga a la neurona biológica es el elemento procesador, PE (*Process Element*). Un elemento procesador tiene varias entradas y las combina, normalmente con una suma básica. La suma de las entradas es modificada por una función de transferencia y el valor de salida de esta función de transferencia pasa directamente a la salida del elemento procesador.

La salida de cada PE puede ir conectada a las entradas de otras neuronas artificiales mediante conexiones ponderadas correspondientes a la eficacia de la sinapsis de las conexiones neuronales.

Si estas unidades elementales de PE están interconectadas entre sí de una manera concreta, forman lo que es llamado como red neuronal. Dependiendo de la conexión entre estos PE los elementos serán unos u otros. Estos PE están organizados en capas, *layers*, estas capas tienen conexiones entre sí.

De cara al trabajo, sólo se trabaja con las capas que están en contacto con el usuario, es decir, las de entrada y salida. En la de entrada se presentan los datos a la red, mientras que la de salida da la respuesta del sistema, lo que viene siendo la clasificación que realiza la red a los datos que le han sido pasados. El resto serán capas ocultas.

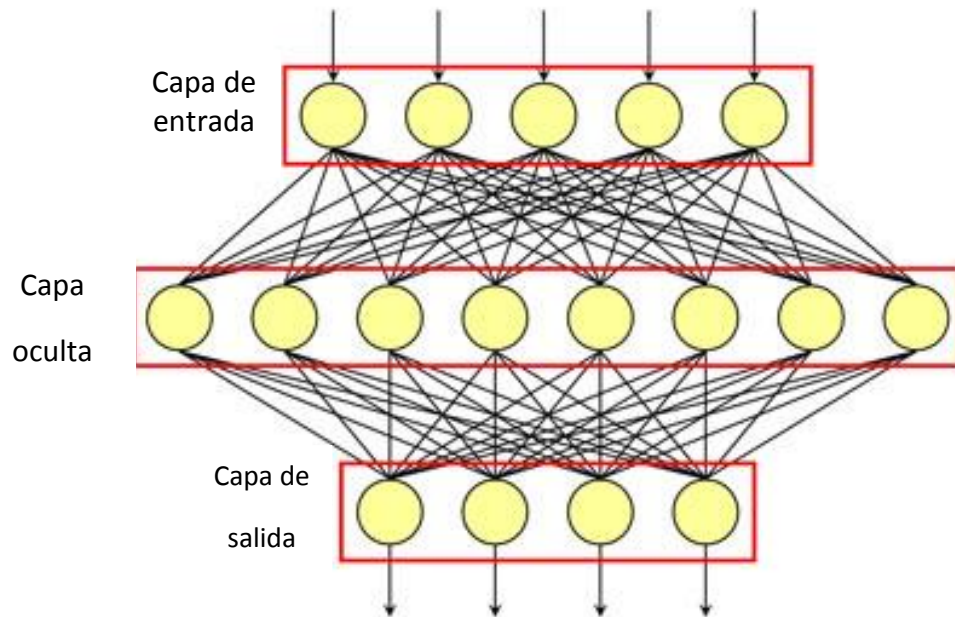


Figura 2. Estructura básica de una red neuronal

#### 1.4 Programación neuronal

Para la solución del problema, se hará uso del programa MATLAB, en él se crean una serie de algoritmos compuestos por un conjunto de instrucciones que indican el modo en el que debe proceder el sistema para lograr solucionar el problema.

Sin embargo, la creación del algoritmo no es suficiente para las ANN, este tipo de redes necesitan ser previamente entrenadas, esto significa que a la red se le muestra en su capa de entrada unos ejemplos y ella misma se ajusta en función de alguna regla de aprendizaje.

Estas redes tienen una serie de limitaciones, por ejemplo, los PE sólo pueden realizar uno, o como mucho, varios cálculos. La potencia del procesamiento de las ANN se mide principalmente por el número de interconexiones actualizadas por segundo durante el proceso de entrenamiento o aprendizaje. La arquitectura de las ANN parte de la organización de los sistemas de procesamiento en paralelo, es decir, sistemas en los que los distintos procesadores están interconectados. No obstante los procesadores son unidades procesadoras simples, diseñadas para la suma de muchas entradas y con un ajuste automático de las conexiones ponderadas.

En el desarrollo de una red neuronal no hay que programar ni el conocimiento ni las reglas de procesamiento del conocimiento. La red neuronal aprende las reglas del procesamiento del conocimiento mediante el

ajuste de las conexiones ponderadas entre las neuronas de distintas capas de la red.

En la computación neuronal las ANN generan sus propias reglas aprendiendo de los ejemplos que se les muestran en la fase de entrenamiento. El aprendizaje se consigue a través una regla de aprendizaje que adapta o cambia los pesos de las conexiones en respuesta a los ejemplos de entrada, y opcionalmente en respuesta a las salidas deseadas. Esta característica de las ANN es lo que permite decir que las redes neuronales aprendan de la experiencia.

Una característica importante de las ANN es la forma o el modo en que se almacena la información. La memoria o el conocimiento de estas redes está distribuida a lo largo de todas las conexiones ponderadas de la red.

Algunas ANN presentan la característica de ser “asociativas” que significa que para una entrada parcial la red elegirá la entrada más parecida en memoria y generará una salida que corresponda a la entrada completa.

La naturaleza de la memoria de las ANN permite que la red responda adecuadamente cuando se le presenta una entrada incompleta o con ruido. Esta propiedad suele ser referida como la capacidad de "generalización".

Otra característica de las ANN es la tolerancia a la falta (*Fault Tolerance*). Tolerancia a la falta se refiere al hecho de que en muchas ANN si resultaran destruidos varios elementos procesadores PE, o se alteraran las conexiones, el comportamiento de la red sería mínimamente modificado. El comportamiento varía pero el sistema no se descompone o deja de funcionar.

Esta característica se debe a que las ANN tienen la información distribuida a lo largo de toda la red.

## 1.5 Historia de la computación neuronal

En 1943, Warren McCulloch y Walter Pitts, neurólogo y estadístico respectivamente, publicaron el artículo “*A logical calculus of Ideas Immanent in Nervous Activity*” [1]. Este artículo supuso la base y el inicio del desarrollo en diferentes campos como son los Ordenadores Digitales, la Inteligencia Artificial y el funcionamiento del ojo.

En 1956, los pioneros de la Inteligencia Artificial, organizaron la primera conferencia sobre este tipo de inteligencia que fue patrocinada por la Fundación Rochester. Esta conferencia fue celebrada en 1956 y en muchos libros se hace referencia a este año como la primera toma de contacto seria con la redes artificiales.

Nathaural Rochester del equipo de investigación de IBM presentó el modelo de una red neuronal que él mismo realizó y puede considerarse como el primer software de simulación de redes neuronales artificiales.

En 1957, Frank Rosenblatt publicó el mayor trabajo de investigación de computación neuronal realizado hasta esas fechas. Su trabajo consistía en el desarrollo de un elemento llamado “Perceptrón” [2]. Este elemento era un sistema capaz de clasificar patrones e identificarlos tanto de tipo geométricos como abstractos. El primer perceptrón era capaz de aprender algo y era robusto, de forma que su comportamiento variaba solo si resultaban dañados los componentes del sistema. Además presentaba la característica de ser flexible y comportarse correctamente después de que algunas celdas fueran destruidas.

El perceptrón estaba compuesto por una rejilla de 400 fotocélulas, correspondientes a las neuronas de la retina sensibles a la luz, recibe el estímulo óptico. Estas fotocélulas están conectadas a elementos asociativos que recogen los impulsos eléctricos emitidos desde las fotocélulas. Las conexiones entre los elementos asociativos y las fotocélulas se realizan de forma aleatoria.

Si las células presentan un valor de entrada superior a un umbral predeterminado entonces el elemento asociativo produce una salida.

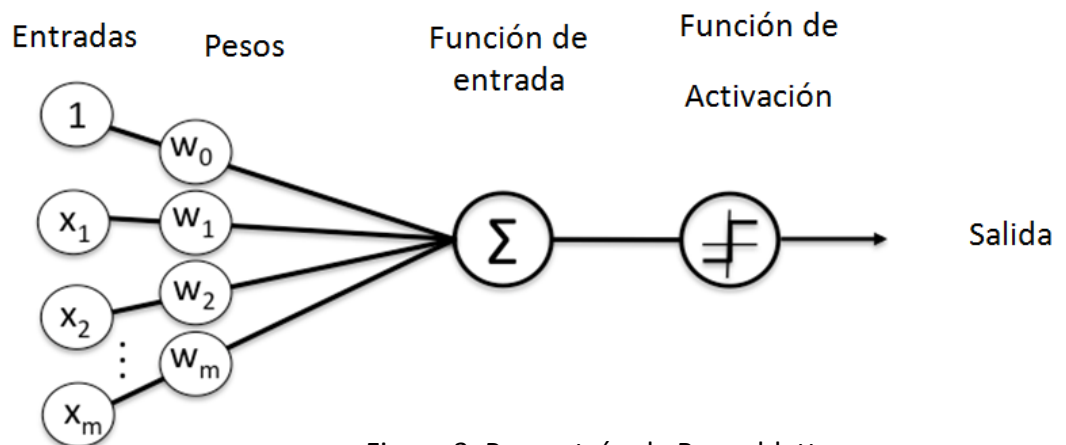


Figura 3. Perceptrón de Rosenblatt

El perceptrón presenta algunas limitaciones debido a que se trataba de un dispositivo en desarrollo. La mayor limitación la reflejaron Minsky y Papert [3] años más tarde, y ponían de manifiesto la incapacidad del perceptrón en resolver algunas tareas o problemas sencillos como por ejemplo la función lógica OR exclusivo.

Uno de los mayores cambios realizados en el perceptrón de Rosenblatt a lo largo de la década de los 60 ha sido el desarrollo de sistemas multicapa que pueden aprender y categorizar datos complejos.

En 1959, Bernard Widrow en Stanford desarrolló un elemento adaptativo lineal llamado "*Adaline*" (*Adaptive Linear Neuron*) [4]. La *Adaline* y una versión de dos capas, llamada "*Madaline*", fueron utilizadas en distintas aplicaciones como reconocimiento de voz y caracteres, predicción del tiempo, control adaptativo y sobre todo en el desarrollo de filtros adaptativos que eliminen los ecos de las líneas telefónicas.

A mediados de los años 60, Minsky y Papert pertenecientes al Laboratorio de Investigación de Electrónica del MIT (Massachusetts Institute Technology) comenzaron un trabajo profundo de crítica al perceptrón. El resultado de este trabajo, el libro "*Perceptrons*", era un análisis matemático del concepto del perceptrón. La conclusión de este trabajo, que se transmitió a la comunidad científica del mundo entero, es que el Perceptrón y la Computación Neuronal no eran temas interesantes a estudiar y desarrollar. A partir de este momento descendieron drásticamente las inversiones en la investigación de la computación neuronal.

Uno de los pocos investigadores que continuaron con su trabajo en la computación neuronal tras la publicación del libro *Perceptrons* fue James Anderson. Su trabajo se basó en el desarrollo de un modelo lineal que consiste en un modelo asociativo-distribuido basado en el principio de Hebb (las conexiones son reforzadas cada vez que son activadas las neuronas) [5].

Una versión extendida de este modelo lineal es el llamado modelo *Brain-State-in-a Box (BSB)*.

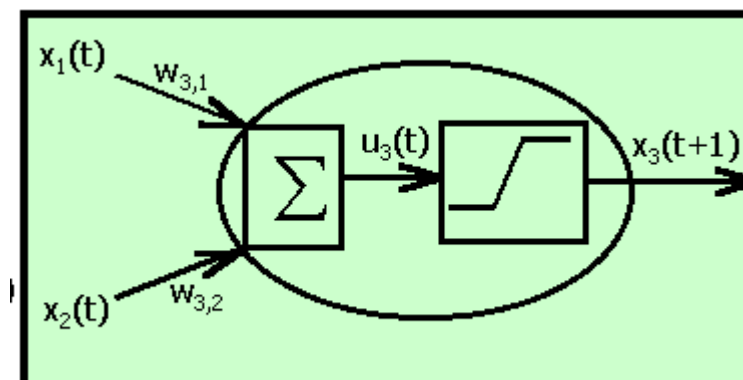


Figura 4. Modelo "*Brain State-in-a Box*"

Teuvo Kohonen [6], de la Universidad de Helsinki, es uno de los mayores impulsores de la computación neuronal de la década de los 70. De su trabajo de investigación destacan dos aportaciones: la primera es la descripción y análisis de una clase grande de reglas adaptativas, reglas en las que las conexiones ponderadas se modifican de una forma dependiente de los valores anteriores y posteriores de las sinapsis. Y la segunda aportación es el principio de aprendizaje competitivo en el que los elementos compiten por

responder a un estímulo de entrada, y el ganador se adapta él mismo para responder con mayor efecto al estímulo.

Otro investigador que continuó con su trabajo de investigación en el mundo de la computación neuronal a pesar del mal presagio que indicaron Minsky y Papert fue Stephen Grossberg. Grossberg estaba especialmente interesado en la utilización de datos de la neurología para construir modelos de computación neuronal. La mayoría de sus reglas y postulados derivaron de estudios fisiológicos. Su trabajo ha constituido un gran impulso en la investigación del diseño y construcción de modelos neuronales. Una de estas clases de redes es la *Adaptive Resonance Theory* (ART) [7].

En 1982 John Hopfield con la publicación del artículo Hopfield *Model o Crossbar Associative Network* [8], junto con la invención del algoritmo retropropagación consiguió devolver el interés y la confianza en el campo de la computación neuronal tras dos décadas de casi absoluta inactividad y desinterés.

Hopfield presenta un sistema de computación neuronal consistente en elementos procesadores interconectados que buscan y tienden a un mínimo de energía. Esta red con este tipo de función de energía y mecanismo de respuesta no es más que un caso de la clase genérica de redes que consideró Grossberg.

## 1.6 Aplicaciones de las redes neuronales artificiales

La gran capacidad de la computación neuronal permite utilizar este tipo de redes en una extensa variedad de aplicaciones. Esta computación permite un mayor reconocimiento y percepción humana que los métodos tradicionales de cálculo.

Algunos de los campos en los que se han aplicado con éxito las ANN son:

- Conversión de texto a voz
- Procesado natural del lenguaje
- Compresión de imágenes
- Reconocimiento de caracteres
- Problemas de combinatoria
- Procesado de señal
  - Predicción
  - Modelado de sistemas
  - Filtrado de ruido
- Modelos económicos y financieros

**Conversión de texto a voz:** Presentado por Sejnowski y Rosenberg, la conversión tex-voz consiste en cambiar los símbolos gráficos de un texto en lenguaje hablado. Este sistema de computación neuronal *NetTalk*, permite convertir texto en fonemas y mediante un sintetizador de voz genera voz a partir de un texto escrito.

La ventaja ofrecida por la computación neuronal es la capacidad de eliminar la necesidad de programar un complejo conjunto de reglas de pronunciación en el ordenador.

**Procesado natural del lenguaje:** Propulsado por los científicos del conocimiento Rumelhart y McClelland, incluía el estudio de cómo se construyen las reglas del lenguaje. La computación neuronal poseía la capacidad de generalizar a partir de datos incompletos así como la capacidad de abstraer.

**Compresión de imágenes:** Consiste en la transformación de los datos de una imagen a una representación diferente que requiera menos memoria.

**Reconocimiento de caracteres:** Proceso de interpretación visual y de clasificación de símbolos. Desarrollado por los investigadores de la empresa Nestor, Inc., se trataba de un sistema de computación neuronal que tras el entrenamiento con un conjunto de tipos de caracteres de letras era capaz de interpretar un tipo de carácter o letra que no haya visto.

**Problemas de combinatoria:** Para solucionar estos tipos de problemas, un ordenador mediante un cálculo tradicional requiere un tiempo de proceso elevado (CPU), que variará lógicamente con el número de entradas. Para resolver el problema del uso excesivo de la CPU se implementan ciertas ANN que cumplan los objetivos.

**Procesado de señal:** Las ANN se han focalizado en tres clases diferentes clases de procesamiento de la señal, como son la predicción, el modelado de un sistema y el filtrado de ruido.

**Predicción:** En la vida real, muchas veces existen fenómenos de los que conocemos su comportamiento a través de una serie temporal de datos o valores. En la laboratorio de Investigación de los Álamos, demostraron que mediante el uso de una red retropropagación (ANN) se conseguían mejores resultados que los obtenidos mediante los métodos de predicción polinómicos y lineales convencionales.

**Modelado de sistemas:** Los sistemas lineales son caracterizados por la función de transferencia que no es más que una expresión analítica entre la variable de salida y una variable independientes y sus derivadas. Las ANN también son capaces de aprender una función de transferencia

y comportarse correctamente como el sistema lineal que está modelando.

**Filtrado de ruido:** Las ANN son capaces de eliminar ruido de una señal. Son capaces de reproducir con gran fidelidad las estructuras y valores de los filtros tradicionales.

**Modelos económicos y financieros:** En esta aplicación se hará un mayor enfoque más adelante, como introducción indicar que consiste en crear modelos y pronósticos económicos, por ejemplo volumen de ventas, intereses de préstamos o volumen de ventas.

## 2. Fundamentos de las redes neuronales artificiales

Las redes neuronales artificiales se basan en el funcionamiento del sistema neuronal del cuerpo humano. En el cuerpo humano encontramos 3 elementos fundamentales:

- **Órganos receptores:** Recogen información del exterior
- **Sistema nervioso:** Transmite la información, la analiza, almacena parte de ella y envía la información elaborada.
- **Órganos efectores:** Reciben la información de parte del sistema nervioso y la convierte en una cierta acción.

La unidad fundamental del sistema nervioso es la neurona, que tienen la estructura explicada anteriormente.

### 2.1 Redes neuronales artificiales

Las redes neuronales artificiales tratan de emular las características y propiedades de las redes neuronales biológicas. En general, consisten en una serie de unidades denominadas neuronas conectadas entre sí.

Cada neurona recibe un valor de entrada, este valor se transforma según una función específica denominada función de activación presente en cada neurona. Dicha señal transformada pasa a ser la salida de la neurona. Las neuronas se conectan entre sí según una determinada arquitectura.

Cada conexión tiene un determinado peso que pondera cada entrada a la neurona. De esta manera la entrada de cada neurona es la suma de las salidas de



las neuronas conectadas a ella multiplicadas por el peso de la respectiva conexión. La figura siguiente ilustra dicho concepto:

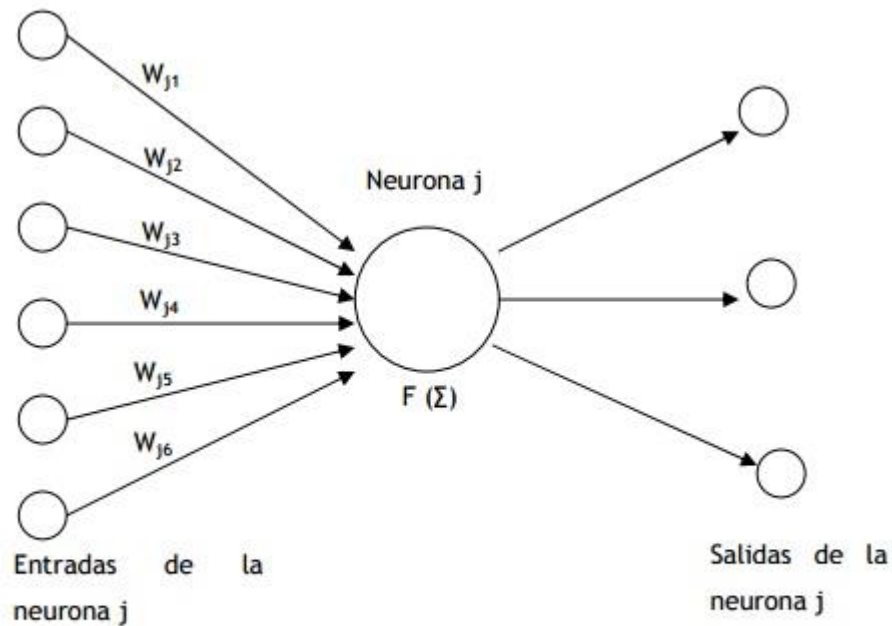


Figura 5. Esquema básico de funcionamiento de una neurona

En este modelo, la neurona  $j$  recibe una serie de entradas  $x_1, x_2, \dots, x_n$ . Cada señal se multiplica por el peso asociado a su conexión,  $w_1, w_2, \dots, w_n$ . Luego, se suman estas entradas ponderadas y se les aplica la función de activación  $F(.)$  para generar la señal de salida de la neurona  $j$ . Los valores de los pesos son ajustados durante la fase de aprendizaje. Como se ha comentado anteriormente, estas neuronas están conectadas entre sí de acuerdo a una determinada arquitectura. Es decir, las neuronas se agrupan en distintas capas: una capa de entrada, otra de salida, y en el caso de existir, una o varias capas ocultas. La salida de cada neurona se propaga por igual por estas conexiones hasta las neuronas de destino. Cada conexión tiene un peso asociado que pondera el valor numérico de la señal que viaja por ésta. Así pues, una red de neuronas artificial puede verse como un grafo cuyos nodos tienen funcionamiento similar, los cuales propagan la información a través de las distintas conexiones.

Veamos el funcionamiento de una red. Para ello nos referimos a la figura 6. Las entradas a la red son introducidas en las neuronas de la capa de entrada, que normalmente genera una salida tal cual o las escala para que las señales se encuentren en un determinado rango. Estas entradas son propagadas a las neuronas de la siguiente capa. De acuerdo a la ecuación siguiente (1) cada neurona  $j$  de la segunda capa generará una salida de valor:

$$S_{2j} = F_{2j}(X_1 W_{1j}) \quad (1)$$

Donde  $X_1$  es el vector de entradas de la capa 1 y  $W_{1j}$  el vector de pesos correspondientes a las conexiones que van de todas las neuronas de la primera capa a la neurona  $j$  de la segunda capa. La función  $F_{2j}$  es la función de activación de la neurona  $j$  de la segunda capa. Así con todas las neuronas de la segunda capa. Estas salidas son propagadas a las neuronas de la capa de salida. Estas neuronas generan las salidas de la red. Cada neurona  $i$  de la capa de salida generará una salida de valor (2):

$$S_{2j} = F_{2j}(W_{2i} S_2) \quad (2)$$

Donde  $W_{2i}$  es el vector de pesos correspondientes a las conexiones que van de las neuronas de la segunda capa a la neurona  $i$  de la capa de salida, y  $S_2$  el vector de salidas de las neuronas de la capa dos, que a su vez son entradas de las neuronas de la capa de salida.

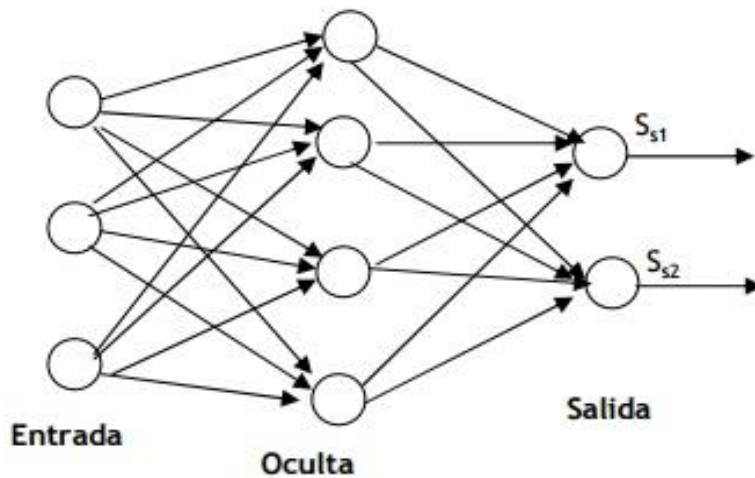


Figura 6. Esquema de la arquitectura de una red

Por último hablar del aspecto más importante y delicado de redes neuronales, el aprendizaje. Las ANN son sistemas de aprendizaje basadas en datos que son utilizados como patrones. Por ello la capacidad de una red de resolver un problema está muy ligada a los patrones utilizados durante su fase de aprendizaje.

El aprendizaje de una red neuronal consiste en hallar los valores precisos de los pesos de sus conexiones para que pueda resolver un determinado problema. El proceso general consiste en ir introduciendo una serie de datos patrón y ajustar los pesos siguiendo un determinado criterio. Los criterios que se van a utilizar en

este proyecto se fundamentan en el error cometido por la red, lo que nos obliga a conocer la salida que se debería obtener para cada uno de ellos.

## 2.2 Tipos de redes neuronales

Existen varios tipos de redes neuronales, la selección de una red u otra depende del problema a resolver. Estas redes pueden ser clasificadas en aplicaciones de Predicción, Clasificación, Asociación, Conceptualización, Filtrado y Optimización. Los tres primeros tipos requieren de un entrenamiento supervisado. A continuación se nombran las más importantes junto con sus respectivas funciones y aplicaciones.

Nombre	Función	Aplicaciones
<b>Adaline y Madaline (Window, 1960)</b>	Predicción	Técnicas de adaptación para el reconocimiento de patrones
<b>Back-Propagation (Rumelhart, 1985)</b>	Clasificación	Solución a las limitaciones de su red predecesora el Perceptron
<b>The Boltzmann Machine (Ackley, 1985)</b>	Asociación	Similar a la red Hopfield
<b>Brain-State-in a Box (Anderson, 1970-86)</b>	Asociación	Red asociativa lineal
<b>Cascade-Correlation-Networks (Fahman, 1990)</b>	Asociación	Adición de nuevas capas ocultas en cascada.
<b>Counter-Propagation (Hecht-Nielsen, 1987)</b>	Clasificación	Clasificación adaptativa de patrones
<b>Delta-Bar-Delta (DBD) Networks (Jacobb, 1988)</b>	Clasificación	Métodos Heurísticos para acelerar la convergencia
<b>Directed Random Search Networks (Maytas, 1965-81)</b>	Clasificación	Técnicas de valores Random en el mecanismo de Ajuste de Pesos
<b>Functional-link Networks (FLN) (Pao, 1989)</b>	Clasificación	Versión mejorada de la red Backpropagation
<b>Learning Vector Quantization (LVQ) Networks (Kohonen)</b>	Optimización	Red clasificadora
<b>Perceptron Networks (Rosenblatt, 1950)</b>	Predicción	Primer modelo de Sistema Neuronal Artificial
<b>Probabilistic Neural Network (PNN)(Spetcht, 1988)</b>	Asociación	Clasificación de patrones utilizando métodos estadísticos
<b>Self-Organizing Maps (SOM) (Kohonen. 1979-1982)</b>	Conceptualización	Aprendizaje sin supervisión

Figura 7. Principales ANN a lo largo de la historia

Esta tabla nos muestra la gran cantidad de redes artificiales neuronales dependiendo de la aplicación. A continuación haré un mayor enfoque en la red de Kohonen puesto que ha sido la red empleada para realizar las pruebas.

### 3. Red Self Organizing Map

El profesor finlandés Teuvo Kohonen [6], presentó en 1982 un modelo de red basado en ciertas evidencias descubiertas a nivel cerebral, denominado mapas auto-organizados o SOM (*Self-Organizing Maps*). Este tipo de red posee un aprendizaje no supervisado competitivo. Esto último quiere decir que no existe un supervisor externo que dé como correcta o incorrecta la operación que haya realizado la red porque no se dispone de ninguna salida objetivo hacia la cual la red neuronal deba tender.

El procedimiento que seguirá la red es intentar descubrir rasgos comunes, regularidades, correlaciones o categorías en los datos de entrada, e incorporarlos a su estructura interna de conexiones. Por tanto, las neuronas deben ser capaces de auto-organizarse en función de los estímulos procedentes del exterior.

Tal y como se ha mencionado anteriormente, el aprendizaje es competitivo, es decir, las neuronas compiten unas con otras con el fin de llevar a cabo una tarea dada. Se pretende que cuando se presente a la red un patrón de entrada, sólo una neurona de salida o grupo de vecinas se active.

Por lo que las neuronas compiten unas con otras por activarse, quedando finalmente una como vencedora y el resto son forzadas a sus valores de respuesta mínimos.

El fin de este aprendizaje es categorizar los datos que se introducen en la red, se clasifican los valores similares en la misma categoría y, por tanto, deben activar la misma neurona.

Las categorías no son establecidas por un supervisor, sino que es la misma red la que las crea a través de correlaciones entre los datos de entrada.

### 3.1 Ideas intuitivas sobre el algoritmo del SOM

- Se trata de un algoritmo capaz de clasificar observaciones
- Se elige un gran número de *clusters* y se colocan en forma de una red bidimensional. La idea es que los representantes de cada grupo (o pesos) estén correlacionados espacialmente, de modo que los puntos más próximos en la rejilla sean más parecidos entre sí que los que estén muy separados.
- El algoritmo es conceptualmente similar al MDS (Análisis de Escalamiento Multidimensional), que transforma observaciones similares en puntos cercanos del espacio bidimensional.
- Si el espacio bidimensional es discretizado mediante su división, por ejemplo, en una rejilla de componentes rectangulares se puede definir una aplicación desde el espacio de alta dimensiones originales sobre dicho espacio bidimensional.
- También se puede tomar la media de los elementos que se encuentran en cada elemento de la rejilla para definir representantes de las clases de la rejilla. Los representantes que están en clases próximas se parecen entre sí. La idea básica del SOM es, así, proporcionar una versión discreta del MDS.

### 3.2 Fundamentos biológicos

Se ha observado que en el córtex de los animales superiores aparecen zonas donde las neuronas detectoras de rasgos se encuentran topológicamente ordenadas; de forma que las informaciones captadas del entorno a través de los órganos sensoriales, se representan internamente en forma de mapas bidimensionales.

Aunque en gran medida esta organización neuronal está predeterminada genéticamente, es probable que parte de ella se origine mediante aprendizaje. Esto sugiere, por tanto, que el cerebro podría poseer la capacidad inherente de formar mapas topológicos a partir de las informaciones recibidas del exterior.

También se ha observado que la influencia que una neurona ejerce sobre las demás en función de la distancia entre ellas, siendo muy pequeña cuando están muy alejadas.

El modelo de red auto-organizado presentado por Kohonen pretende mimetizar de forma simplificada la capacidad del cerebro de formar mapas topológicos a partir de las señales recibidas del exterior.

### 3.3 Arquitectura del SOM

Un modelo SOM está compuesto por dos capas de neuronas. La capa de entrada (formada por  $N$  neuronas, una por cada variable de entrada) se encarga de recibir y transmitir a la capa de salida la información procedente del exterior. La capa de salida (formada por  $M$  neuronas) es la encargada de procesar la información y formar el mapa de rasgos. Normalmente, las neuronas de la capa de salida se organizan en forma de mapa bidimensional como el que sigue a continuación:

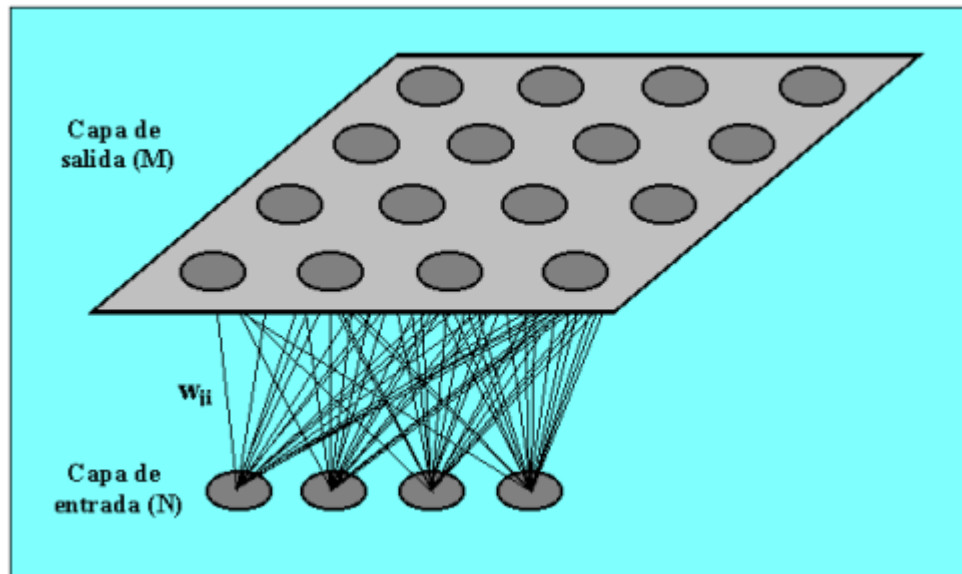


Figura 8. Estructura modelo SOM

Las conexiones entre las dos capas son siempre hacia adelante, es decir, la información siempre se propaga desde la capa de entrada hacia la capa de salida. Cada neurona de entrada  $i$  está conectada con una de las neuronas  $j$  mediante el peso  $w_{ij}$ . De esta forma, las neuronas de salida tienen asociado un vector de pesos  $W_j$  llamado vector de referencia, debido a que constituye el vector prototipo de la categoría representada por la neurona de salida  $j$ . Así, el SOM define una proyección desde un espacio de datos en alta dimensión a un mapa bidimensional de neuronas.

Entre las neuronas de la capa de salida, puede decirse que existen conexiones laterales de excitación o inhibición implícitas, pues aunque no estén conectadas, cada una de estas neuronas va a tener influencia sobre sus vecinas. Esto se consigue a través de un proceso de competición entre las neuronas y de la aplicación de una función denominada de vecindad, que produce la topología o estructura del mapa. Las topologías más frecuentes son la rectangular y la hexagonal.

Las neuronas adyacentes pertenecen a una vecindad  $N_j$  de la neurona  $j$ . La topología y el número de neuronas permanecen fijos desde el principio. El

número de neuronas determina la suavidad de la proyección, lo cual influye en el ajuste y capacidad de generalización del SOM. Durante la fase de entrenamiento, el SOM forma una red elástica que se pliega dentro de la nube de datos originales. El algoritmo controla la red de modo que tiende a aproximar la densidad de los datos. Los vectores de referencia del *codebook* (vectores de pesos que conforman la red en cuestión) se acercan a las áreas donde la densidad de datos es alta. Eventualmente unos pocos vectores del *codebook* están en áreas donde existe baja densidad de datos.

### 3.4 El algoritmo del SOM

El proceso de aprendizaje del SOM es el siguiente:

**Paso 1.** Un vector  $x$  es seleccionado al azar del conjunto de datos y se calcula su distancia (similitud) a los vectores del *codebook*, usando, por ejemplo, la distancia euclídea (3):

$$\|x - m_c\| = \min_j \{\|x - m_j\|\} \quad (3)$$

**Paso 2.** Una vez que se ha encontrado el vector más próximo o BMU (*Best Matching Unit*) el resto de vectores del *codebook* es actualizado. El BMU y sus vecinos (en sentido topológico) se mueven cerca del vector  $x$  en el espacio de datos. La magnitud de dicha atracción está regida por la tasa de aprendizaje. Mientras se va produciendo el proceso de actualización y nuevos vectores se asignan al mapa, la tasa de aprendizaje decrece gradualmente hacia cero. Junto con ella también decrece el radio de vecindad también. La regla de actualización para el vector de referencia dado  $i$  es la siguiente (4):

$$m_j(t+1) = \begin{cases} m_j(t) + \alpha(t)(x(t) - m_j(t)) & j \in N_c(t) \\ m_j(t) & j \notin N_c(t) \end{cases} \quad (4)$$

Los pasos 1 y 2 se van repitiendo hasta que el entrenamiento termina. El número de pasos de entrenamiento se debe fijar antes a priori, para calcular la tasa de convergencia de la función de vecindad y de la tasa de aprendizaje. Una vez terminado el entrenamiento, el mapa ha de ordenarse en sentido topológico:  $n$  vectores topológicamente próximos se aplican en  $n$  neuronas adyacentes o incluso en la misma neurona.

## 4. Business Intelligence

### 4.1 Introducción

Alrededor de los años 80, las empresas almacenaban enormes cantidades de datos relacionados con las actividades de las mismas, sin embargo, las aplicaciones de gestión de estos datos estaban prácticamente obsoletas. Es por esta razón por la que a partir de estos años comienzan a aparecer soluciones destinadas a solucionar estos problemas existentes con esos sistemas de gestión de datos.

Con el paso de los años fueron apareciendo una serie de técnicas, herramientas y metodologías que, de forma conjunta, conforman lo que se conoce como el término *Business Intelligence*, acuñado por Howard Dresner [10] del grupo Gartner en 1989. Este término pretende ser la base para reunir a todo tipo de tecnologías capaces de extraer los datos corporativos almacenados por un sistema de gestión y tratarlos de manera que, al presentárselos a cualquier persona, pueda obtener un conocimiento intelectual para así llevar a cabo las tareas necesarias para la consecución exitosa de las metas propuestas en su negocio.

### 4.2 ¿Qué es Business Intelligence?

A través de una serie de definiciones quedará definido que es *Business Intelligence* o Inteligencia de Negocio:

*“Llamamos Business Intelligence (BI) al conjunto de estrategias y herramientas enfocadas a la administración y creación de conocimiento mediante el análisis de datos existentes en una organización o empresa.”*  
(Wikipedia)

*“La inteligencia de negocio es un proceso sistemático de recolección, análisis y gestión de información interna y externa y de conocimiento para mejorar el proceso de toma de decisiones de una empresa.”*  
(Jay Liebowitz, *Strategic Intelligence: Business Intelligence, Competitive Intelligence, and Knowledge Management*, 2006)

*“Conjunto de tecnologías, métricas, procesos y sistemas que una organización usa para controlar y gestionar su rendimiento empresarial.”*  
(Bani Brandolini, *Presidente Internacional de Tagetik*)

*“Business Intelligence (BI) es un conjunto de conceptos y metodologías para mejorar la toma de decisiones a través del uso de hechos y sistemas basados en hechos.”*  
(Gartner Group)



*“BI es un proceso interactivo para explorar y analizar información estructurada sobre un área (normalmente almacenada en un datawarehouse), para descubrir tendencias o patrones, a partir de los cuales derivar ideas y extraer conclusiones”.*  
*El proceso de Business Intelligence incluye la comunicación de los descubrimientos y efectuar los cambios.*  
*Las áreas incluyen clientes, proveedores, productos, servicios y competidores.”*

Esta última definición puede ser descompuesta como se muestra a continuación:

- **Proceso interactivo:** Se trata de un análisis de información continuado en el tiempo.
- **Explorar:** En todo proyecto de inteligencia de negocio existe un momento inicial en el que por primera vez accedemos a información que nos facilita su interpretación. En esta primera fase lo que hacemos es “explorar” para comprender qué sucede en nuestro negocio.
- **Analizar:** De lo que se trata es de descubrir relaciones entre variables, tendencias o patrones.
- **Información estructurada y datawarehouse:** La información que utilizamos en la inteligencia de negocio se encuentra almacenada en tablas relacionadas entre ellas. Las tablas tienen registros y cada uno de los registros tienen distintos valores para cada uno de los atributos. Estas tablas están almacenadas en lo que conocemos como *datawarehouse* o almacén de datos, que simplemente contiene una colección de datos orientados a un determinado ámbito.
- **Área de análisis:** En todo proceso de este tipo existe un objeto de análisis concreto.
- **Comunicar los resultados y efectuar los cambios:** Una vez encontrado los patrones y sacadas conclusiones, hay que comunicarlo a aquellas personas que tengan que realizar los cambios pertinentes en la organización para mejorar nuestra competitividad.

Definir que en este trabajo, se explica en conjunto de operaciones que lleva consigo un proceso de *Business Intelligence*, sin embargo, para realizar los experimentos me he centrado en la categorización de los datos así como la obtención de patrones de clasificación.

### 4.3 Beneficios que aporta el *Business Intelligence*

El objetivo básico de este tipo de proceso es obtener información de cara a la toma de decisiones y adquirir mayor conocimiento que nos sea útil en un futuro.

Se puede distinguir una serie de tipos de beneficios:

- **Beneficios tangibles:** En esta categoría se encuentran los relacionados con la reducción de costes, generación de ingresos o reducción de tiempos en la actividad del negocio.
- **Beneficios intangibles:** Al tener a nuestro alcance mayor información disponible para la toma de decisiones hará que más usuarios utilicen dicha información para tomar decisiones y mejorar nuestra posición competitiva.
- **Beneficios estratégicos:** Bajo esta categoría se encuentran todo aquel beneficio que nos facilita la formulación de la estrategia, es decir, a qué clientes, mercados o con qué productos dirigirnos.

### 4.4 Componentes *Business Intelligence*

Para comenzar cualquier tipo de proyecto debemos conocer los objetivos, el alcance de la solución y qué modelos de negocio queremos analizar. Con esta información es más sencillo tomar las decisiones necesarias en cada uno de los componentes.

Los componentes principales se pueden ver en el siguiente gráfico:

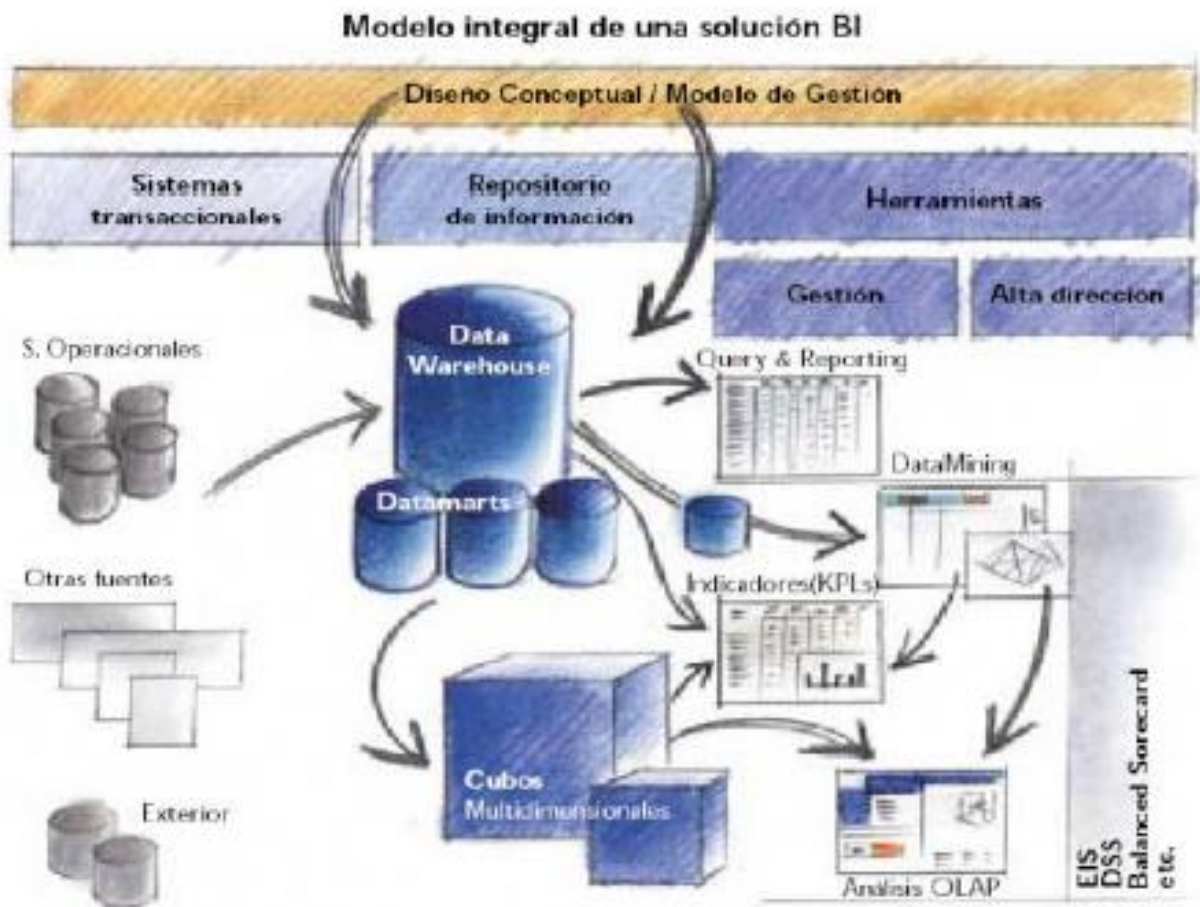


Figura 9. Componentes principales en un proceso de *Business Intelligence*

- **Fuentes de información**, de las cuales partiremos para alimentar de contenidos el *datawarehouse*.
- **Proceso ETL** de extracción, transformación y carga de datos en el *datawarehouse*. Antes de almacenar los datos en un *datawarehouse*, estos deben ser transformados, limpiados, filtrados y redefinidos.
- El propio ***datawarehouse*** o almacén de datos.
- El motor **OLAP (*On-Line Analytical Processing*)**, que nos debe proveer capacidad de cálculo, consultas, funciones de planeamiento, pronóstico y análisis de escenarios en grandes volúmenes de datos.
- Las **herramientas de visualización**, que nos permitirán el análisis y la navegación a través de los mismos.

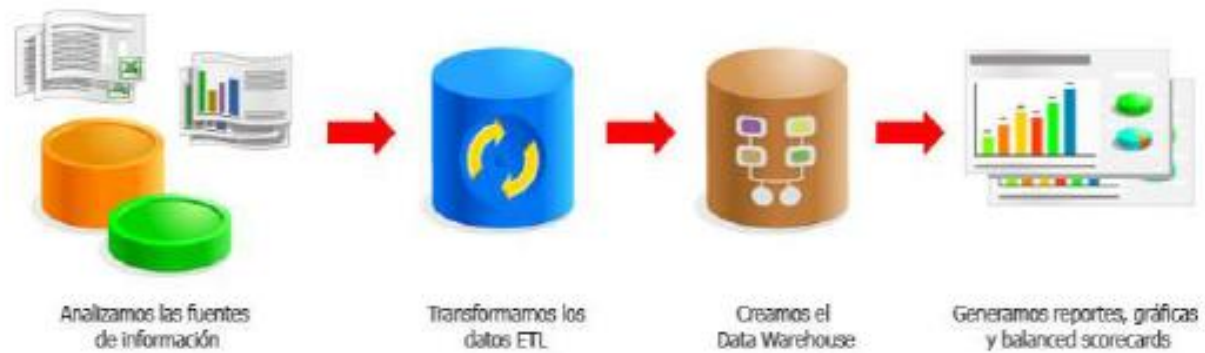


Figura 10. Esquema *Business Intelligence*

#### 4.4.1 Fuentes de información

Vamos analizar las distintas fuentes de información con las que podemos alimentar un *datawarehouse*.

Las fuentes de información a las que podemos acceder son:

- Básicamente, de los sistemas operacionales o transaccionales, que incluyen aplicaciones desarrolladas a medida, ERP (Software de gestión de recursos empresariales), CRM (Administración basada en la relación basada con los clientes), SCM (Administración de redes de suministro), etc.
- Sistemas de información departamentales: previsiones, presupuestos, hojas de cálculo, etc.
- Fuentes de información externa, en algunos casos comprada a terceros, como por ejemplo estudios de mercado.



Figura 11. Fuentes de información principales proceso BI

Existen muchos factores que contribuyen a la complejidad de cargar la información en un *datawarehouse*. Uno de los principales es el número de fuentes de información distintas de las que cargamos la información.

Acceder a distintas bases de datos requiere distintas habilidades y el conocimiento de distintas sintaxis de SQL. Si el número de bases de datos a las que debemos acceder es elevado, puede provocar que tanto las definiciones como las codificaciones en los distintos entornos sean diferentes, lo que añadirá dificultad a nuestro proyecto.

La información que cargamos en un *datawarehouse* normalmente es estructurada, es decir, aquella que se puede almacenar en tablas: en la mayoría de los casos es información numérica.

Tendremos que analizar si la información de la que disponemos es la que necesitamos para alimentar los modelos de negocio que hemos definido anteriormente.

Una vez decididas las fuentes de información debemos verificar la calidad de los datos.

#### 4.4.2 Proceso de extracción, transformación y carga (ETL)

Este proceso nos permitirá alimentar un *datawarehouse*. El proceso trata de recuperar los datos de las fuentes de información y alimentar el *datawarehouse*.

El proceso de ETL consume entre el 60% y el 80% del tiempo de un proyecto de *Business Intelligence*, por lo que es un proceso clave en la vida de todo proyecto.

El proceso ETL se divide en 5 subprocesos:

- **Extracción:** Este proceso recupera los datos físicamente de las distintas fuentes de información. En este momento disponemos de los datos en bruto.
- **Limpieza:** Este proceso recupera los datos en bruto y comprueba su calidad, elimina los duplicados y, cuando es posible, corrige los valores erróneos y completa los valores vacíos, es decir, se transforman los datos -siempre que sea posible- para reducir los errores de carga. En este momento disponemos de datos limpios y de alta calidad.
- **Transformación:** Este proceso recupera los datos limpios y de alta calidad y los estructura en los distintos modelos de análisis. El resultado de este proceso es la obtención de datos limpios, consistentes y útiles.
- **Integración:** Este proceso verifica que los datos que cargamos en el *datawarehouse* son consistentes con las definiciones y formatos del *datawarehouse*; los integra en los distintos modelos de las distintas áreas de negocio que hemos definido en el mismo. Estos procesos pueden ser complejos.
- **Actualización:** Este proceso es el que nos permite añadir los nuevos datos al *datawarehouse*.

#### 4.4.3 Datawarehouse o almacén de datos

Según Hugh J. Watson un *datawarehouse* es una colección de información creada para soportar las aplicaciones de toma de decisiones. El *datawarehouse* nos permitirá analizar la información contextualmente y relacionada dentro de la organización.

Este *datawarehouse* está construida para resolver un problema de negocio, que ha sido definido por los que están autorizados para la toma de decisiones. La información aquí existente debe ser transformada en medidas comunes, códigos comunes y formatos comunes para que pueda ser útil. Esta información deber almacenada referida a unidades de tiempo para tener claro de qué momento es cada dato.

#### 4.4.4 Herramientas y técnicas

Existen una gran cantidad de herramientas y técnicas, relacionadas entre sí o independientes, la utilización de una o de otras depende de varios factores, como los medios físicos disponibles, los datos de entrada que se posean o la salida que pretendamos obtener. A continuación se detallan los más importantes, el empleado en las pruebas son las redes neuronales pertenecientes a las herramientas de minería de datos.

### **OLTP (*On-line Transaction Processing*)**

Tecnología que se utiliza para administrar aplicaciones que utilizan operaciones transaccionales, es decir, sistemas donde se realizan una gran cantidad de modificaciones y entradas de datos y pocas lecturas masivas de los mismos. En estos sistemas es necesario tener un tiempo de respuesta aceptable a la hora de realizar las modificaciones de los datos.

### **OLAP (*On-line Analytical Processing*)**

Estas herramientas manejan una serie de consultas de forma interactiva sobre estructuras multidimensionales (Cubos OLAP) cargadas previamente con los datos almacenados en las bases de datos corporativas tradicionales. Permiten realizar informes y obtener grandes cantidades de información a partir de lo que resultaría ser a modo rutinario una serie de complejas consultas sobre una base de datos de forma sencilla.

Con estos sistemas es posible analizar la información almacenada en un *datawarehouse*, pero no es estrictamente necesario, ya que la información puede provenir de diferentes bases de datos. El objetivo de estas herramientas es obtener una mejor comprensión de lo almacenado en las bases de datos.

### ***Query&Reporting***

Herramientas para elaborar informes y listados no demasiado complejos con los datos más usuales y sencillos de analizar, tanto de manera agregada como detallada de la información. Este tipo de herramientas trabajan de forma óptima a través de sistemas basados en almacenes de datos, ya que su tecnología y su forma de estructurar la información favorecen que con estas herramientas se obtengan tiempos de respuesta menores en las consultas, no siendo así en los sistemas tradicionales.

Estas herramientas ofrecen diferentes formas de presentación y diseño (Excel, PDF,...), según el tipo de información analizada y dependiendo de para qué causa esté destinada. Algunas de estas herramientas ofrecen al usuario la posibilidad de refrescar la información de informes almacenados y que poseen información estática por medio de alarmas previamente programadas. De esta manera, el usuario siempre dispondrá de la información actualizada.

### **Cuadro de mando integral**

El Cuadro de Mando Integral (CMI) o *Balanced Scorecard* fue presentado por los consultores e investigadores de negocios Robert Kaplan y David Norton en 1992 como base de un trabajo realizado para una empresa de semiconductores.

El Cuadro de Mando toma como núcleo principal la visión estratégica y la organización para aplicarla en diferentes perspectivas a toda la empresa y así mejorar su rendimiento por medio de indicadores de acción, objetivos y estándares. Estas perspectivas son:

- **Perspectiva financiera:** Esta perspectiva aborda los objetivos financieros de la empresa. Consiste en ofrecer una estrategia sobre la contabilidad por medio de indicadores financieros para obtener un estado financiero favorable para la empresa.
- **Perspectiva del cliente:** Para obtener el estado financiero deseado se necesitan estrategias para obtener clientes y satisfacer sus necesidades. Gracias a esta perspectiva se miden los indicadores y se tratan los objetivos que ayudan a la empresa en la relación con el cliente.
- **Perspectiva de procesos:** Desde esta perspectiva se pretende controlar las actividades internas de la empresa (compras, ventas, producción...etc.) para satisfacer los objetivos propuestos con el resto de categorías.
- **Perspectiva de formación y crecimiento:** Estos indicadores forman el conjunto de recursos humanos que dotan a la organización de habilidades para mejorar y aprender.

A partir de esta serie de categorías en las que aplicar la estrategia de negocio se puede deducir que los cuadros de mando integrales son una potente herramienta de inteligencia de negocios para la dirección donde poder apoyarse y tomar las decisiones estratégicas necesarias para obtener el máximo rendimiento y evolución posible de su empresa.

### ***Data Mining o Minería de datos***

La minería de datos consiste en extraer conocimiento útil a partir de los datos en bruto de una organización. Las empresas almacenan grandes cantidades de información oculta en sus datos y gracias a estas técnicas y herramientas informáticas y estadísticas es posible que esta información vea la luz, aportando conocimiento beneficioso para el usuario por medio de clasificaciones y predicciones.

El objetivo de estas técnicas no es otro que encontrar patrones ocultos de comportamiento, tendencias y correlaciones entre los datos para disponer de suficiente información como para realizar modelos estadísticos que pueden servir para prever ciertas situaciones de la organización. Esta serie de patrones y tendencias se suelen agrupar en lo que se denomina como



“Modelo de minería”, los cuales se pueden utilizar posteriormente en diferentes escenarios hipotéticos o simulados de negocio. La minería de datos se apoya en una serie de técnicas de adquisición de conocimiento y aprendizaje basadas en:

- Redes neuronales.
- Árboles de decisión.
- Algoritmos genéticos.
- Método del vecino más cercano.
- Reglas de inducción.
- Análisis de series temporales.

Dentro de este tipo de herramientas he hecho uso de la red neuronal de Kohonen.

## 5. Aplicación de la red de Kohonen a *Credit Scoring*

### 5.1 ¿Qué es *Credit Scoring*?

Un *credit scoring* es un sistema de calificación de créditos que intenta automatizar la toma de decisiones en cuanto a conceder o no una determinada operación de riesgo, normalmente un crédito. La virtud de este sistema es la de acortar el tiempo de análisis además de simplificarlo, lo que contribuye a mejorar el nivel de servicios proporcionados a la clientela.

Un ejemplo de esto sería una aplicación de *credit scoring* instalada en un cajero automático, el sistema sería capaz de darle respuesta positiva o negativa a la petición de un cliente de solicitar un crédito. No haría falta que el cliente acudiera a una entrevista personal o acordara una cita con antelación con su banco.

## 5.2 Ejemplo de aplicación

A continuación, como ejemplo paradigmático de la inteligencia artificial aplicada a un caso de negocio, se aplica una red neuronal no supervisada a al problema de concesión automática de créditos (*Credit Scoring*).

*Credit scoring* es una tarea de clasificación binaria, siendo posible dos tipos de resultados: SI o NO. A lo largo de los años han surgido numerosas aplicaciones que han ido aumentando la precisión de los resultados. En la actualidad los mejores resultados son los obtenidos mediante el uso de redes neuronales artificiales (ANN).

Los atributos de los conjuntos con los que se ha trabajado han sido modificados a datos que carecen de sentido para evitar problemas de confidencialidad, los datos que proceden de bancos relacionados con este campo de crédito pueden ser de este tipo, donde se observan que los campos pueden ser tanto numéricos como semánticos:

SI-Fraude-SPI	11/01/2016	31	25	14/11/2015	14/11/2015	3600	NO	DEUDOR	Fraude por infección de inyección	0	3600
SI-Fraude-SPI	11/01/2016	31	25	17/11/2015	17/11/2015	1993	NO	DEUDOR	Fraude por infección de inyección	0	1993
SI-Fraude-SPI	11/01/2016	31	25	18/11/2015	18/11/2015	450	NO	DEUDOR	Fraude por infección de inyección	0	0
SI-Fraude-SPI	11/01/2016	31	25	19/11/2015	19/11/2015	3146	NO	DEUDOR	Fraude por infección de inyección	0	3146
SI-Fraude-SPI	11/01/2016	31	25	20/11/2015	20/11/2015	4950	NO	DEUDOR	Fraude por infección de inyección	0	4950
SI-Fraude-SPI	11/01/2016	31	25	30/10/2015	30/10/2015	5600	NO	DEUDOR	Fraude por infección de inyección	0	5600
SI-Fraude-SPI	11/01/2016	31	25	02/03/2015	02/03/2015	4800	NO	DEUDOR	Fraude por infección de inyección	0	4800
SI-Fraude-SPI	11/01/2016	31	25	02/03/2015	02/03/2015	4000	NO	DEUDOR	Fraude por infección de inyección	0	4000
SI-Fraude-SPI	11/01/2016	31	25	02/03/2015	02/03/2015	5960	NO	DEUDOR	Fraude por infección de inyección	0	5960

Figura 12. Datos bancarios no normalizados

Para obtener únicamente datos de tipo numérico compatibles con nuestra red de Kohonen es necesario aplicarle un proceso de normalización a los datos como el explicado a continuación.

Si los datos de entrada únicamente pueden tomar dos valores, por ejemplo hombre o mujer, se normalizaría de la siguiente manera:

Hombre → 1

Mujer → 2

Conviene resaltar que el valor 0 es mejor no emplearlo para una correcta simulación de los pesos.

Otro factor importante es que los valores de entrada deben estar escalados de forma que todas las magnitudes sean similares para así poder conseguir una mayor eficiencia en el proceso de clasificación.

Asimismo, conviene resaltar que también en la realidad se emplean algoritmos de normalización basados en la media o varianza entre otra, en este aspecto no he entrado puesto que el conjunto de datos venía ya normalizado.

Este proceso de codificar los datos y normalizarlos suele ser referido a él como estandarización. A pesar de que no se le suele dar importancia, es de gran relevancia para una buena clasificación de acuerdo a los pesos.

En teoría no es necesario normalizar los datos numéricos, la red funcionaría correctamente con los datos en bruto. Sin embargo, en la práctica se ha demostrado que una normalización conlleva un mejor entrenamiento de la red, y esto a su vez conlleva a una mejor predicción.

Esto es básicamente debido a lo siguiente, si los datos no son normalizados, entonces un cambio en un peso de la red tiene mayor influencia en el valor de magnitud más alta.

Por ejemplo, el cambio de peso es de 0.1, entonces si tuviéramos valores de 30 y 38000, esos valores variarían respectivamente 0.3 y 3800.

Los dos conjuntos sobre los que se ha tratado son los datos basados en los créditos australiano y alemán.

### **Caso australiano**

El conjunto australiano presenta 307 buenos solicitantes y 383 malos solicitantes. Cada solicitante contiene 15 características, de ellas 6 son nominales y 8 numéricas. La última característica es el tipo de clase, solicitante bueno o malo de cara al riesgo de concederle un crédito.

Creando la red neuronal con únicamente dos neuronas de salida los resultados no se acercaban ni al 20% de acierto, por lo que se fue aumentando en un factor 2 las neuronas de salida y valorando la salida. Obteniendo la mejor salida para 8 neuronas de salida.

Con los siguientes comandos crearía la red el programa y la entrenaría:

```
x = australia'; % Entradas
net = selforgmap([8]); % Crea la red

net = train(net,x); % Entrena la red, proceso de
entrenamiento

view(net) %Visulizar la red gráficamente
```

```

y = net(x); % Salida

classes = vec2ind(y);

```

Como se puede observar, esta red posee 14 neuronas en la capa de entrada y 8 neuronas en la capa de salida.

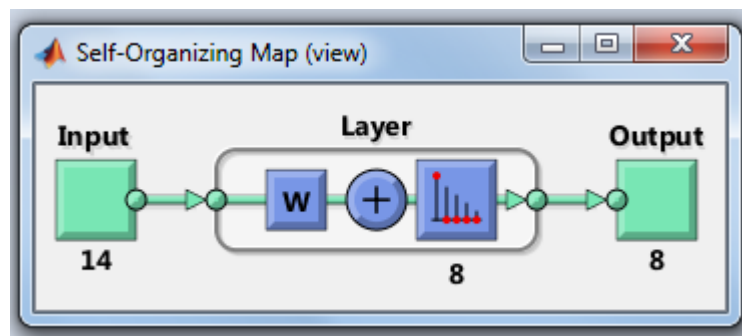


Figura 13. Esquema de la red neuronal compuesta por 14 entradas y 8 salidas.

En la siguiente figura se pueden observar el número de veces que cada neurona se activa, siendo las neuronas números 2, 4 y 6 las salidas que indican que a dicho cliente se le podría conceder un crédito, mientras que el resto de neuronas indicarían que se trata de un cliente "malo". La tasa de acierto ha sido del 79 %, pero del 19%.

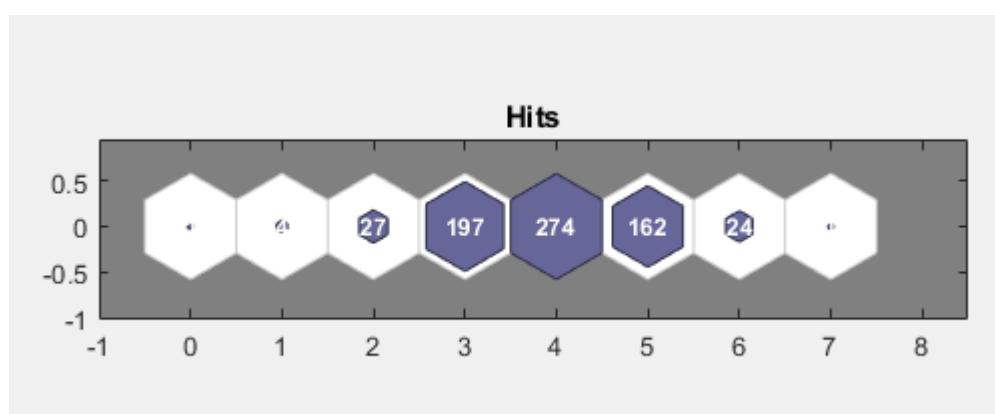


Figura 14. Salidas de la red neuronal para el caso de Australia

### Caso alemán

Este conjunto posee 24 atributos que incluyen los siguientes, tanto numéricos como cualitativos:

- El estatus de la cuenta bancaria
  - 1 → < 0 Decenas de millar (DM)
  - 2 → 0 < ... < 200 DM
  - 3 → ... ≥ 200 DM
  - 4 → No tiene cuenta bancaria
- Duración en meses
- Historial de créditos
  - 0 → Nunca ha pedido un crédito / Todos los créditos pagados a tiempo
  - 1 → Todos los créditos de este banco pagados correctamente
  - 2 → Créditos existentes hasta el momento pagados correctamente
  - 3 → Retraso en pago en algún crédito del pasado
  - 4 → Cuenta crítica / Posee otros créditos en el momento ( Otros bancos)
- Propósito del crédito
  - 0 → Coche nuevo
  - 1 → Coche usado
  - 2 → Muebles
  - 3 → Televisión / Radio
  - 4 → Electrodomésticos
  - 5 → Reformas
  - 6 → Educación
  - 7 → Vacaciones
  - 8 → Recibir algún tipo de formación
  - 9 → Negocios
  - 10 → Otros
- Cantidad solicitada en el crédito
- Cuenta bancaria de ahorro
  - 1 → ... < 100 DM
  - 2 → 100 ≤ ... < 500 DM
  - 3 → 500 ≤ ... < 1000 DM
  - 4 → ... ≥ 1000 DM
  - 5 → Desconocido / No posee cuenta
- Situación laboral actual

- 1→ Desempleado
  - 2→ ... <1 año
  - 3→ 1<=...< 4 años
  - 4→ 4 años <= ...< 7 años
  - 5→ ...>= 7 años
- Porcentaje de la renta disponible para entregar
- Situación personal y sexo
  - 1→ Hombre: Divorciado/Separado
  - 2→ Mujer: Divorciada/Separada/Casada
  - 3→ Hombre: Soltero
  - 4→ Hombre: Casado/Viudo
  - 5→ Mujer: Soltera
- Otros deudores
  - 1→ Ninguno
  - 2→ Co-solicitante
  - 3→ Garante
- Residencia actual
- Propiedades
  - 1→ Propiedades relacionadas con viviendas
  - 2→ Seguro de vida
  - 3→ Coche
  - 4→ Desconocidas / Ninguna propiedad
- Edad
- Otros planes
  - 1→ Banco
  - 2→ Tiendas
  - 3→ Ninguno
- Casa
  - 1→ Alquilada
  - 2→ Propia
  - 3→ Sin gasto
- Número de créditos existentes en el banco

- Trabajo
  - 1 → Desempleado/ Sin formación – No residente
  - 2 → Sin formación -- Residente
  - 3 → Empleado con formación
  - 4 → Empleado altamente cualificado / Oficial
- Número de personas a mantener
- Teléfono
  - 1 → Ninguno
  - 2 → Si, registrado bajo el nombre del cliente
- Trabajador extranjero
  - 1 → Sí
  - 2 → No

Los datos tienen la siguiente forma:

1	6	4	12	5	5	3	4	1	67	3	2	1	2	1	0	0	1	0	0	1	0	0	1
2	48	2	60	1	3	2	2	1	22	3	1	1	1	1	0	0	1	0	0	1	0	0	1
4	12	4	21	1	4	3	3	1	49	3	1	2	1	1	0	0	1	0	0	1	0	1	0
1	42	2	79	1	4	3	4	2	45	3	1	2	1	1	0	0	0	0	0	0	0	0	1
1	24	3	49	1	3	3	4	4	53	3	2	2	1	1	1	0	1	0	0	0	0	0	1

Figura 15. Datos de entrada para el caso de Alemania

Después de numerosas pruebas desde 2 neuronas de salida hasta 20 neuronas, los mejores resultados fueron obtenidos con 6 neuronas de salida

La red sería creada de la siguiente forma:

```
x = alemania'; % Entradas
net = selforgmap([6]); % Crea la red

net = train(net,x); % Entrena la red, proceso de
entrenamiento
```

```

view(net) %Visulizar la red gráficamente

y = net(x); % Salida

classes = vec2ind(y);

```

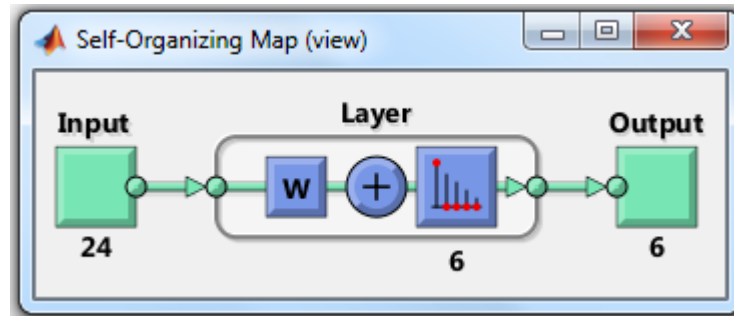


Figura 16. Esquema de la red compuesta por 24 entradas y 6 salidas

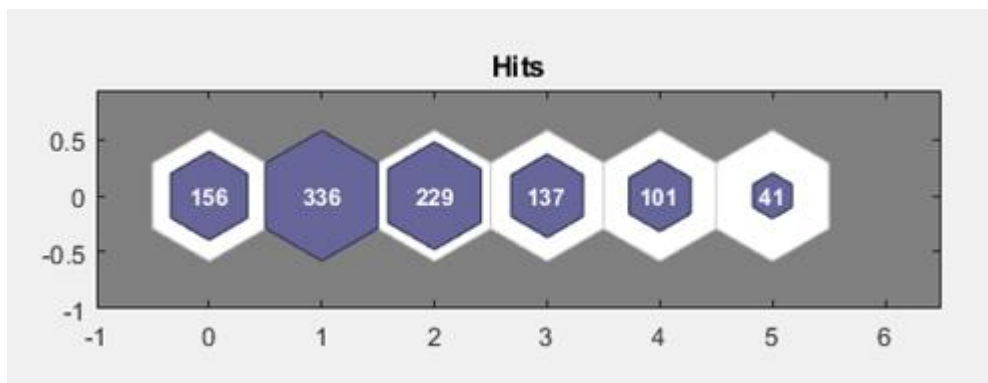


Figura 17. Salidas de la red neuronal para el caso de Alemania

Las neuronas de salida que reunían a los clientes “buenos” son las neuronas número 0, 2 y 3. La tasa de acierto era del 70% mientras que el fallo rondaba el 10%. Asimismo, una vez realizada la categorización y analizando los atributos de entrada se podían observar lo siguiente:

- Si el cliente no posee una cuenta bancaria, automáticamente se le clasificaba como cliente “malo”.
- Si el cliente poseía una cuenta crítica entorno a su historial de créditos, automáticamente se le clasificaba como cliente “malo”.
- Si el cliente posee retrasos en sus pagos, el programa lo clasifica como cliente “malo” en un 70% de las veces.
- Si el cliente es extranjero, la red lo clasifica como cliente “malo” entorno al 65% de las veces.
- Si el cliente se encuentra desempleado, y la cuenta bancaria posee menos de 500DM es clasificado como cliente “malo”.



Analizando los resultados y obviando los errores puede verse que el programa es capaz de clasificar los diferentes casos y categorizarlos según sus atributos.

## 6. Discusión

Una de las líneas de mejora sería utilizar nuevos modelos de Redes Neuronales avanzadas que tratan de incorporar nuevas características bioinspiradas para obtener un nivel de conocimiento más profundo de los datos (Deep Learning) como puede ser la red Koniocortex (KLN), diseñada por Francisco Javier Roperio Peláez y Diego Andina de la Fuente [9]], esta red será explicada a continuación.

Esta red está basada en la koniocortex de los cerebros biológicos, también llamada corteza granular, que son las diferentes regiones de la corteza cerebral que poseen una capa granular interna bien definida (capa IV). Ambos nombres (koniocortex y granulares) se refieren a una corteza que tiene una textura granulada (Konia es una palabra griega que significa "polvo") debido a la abundancia de las neuronas espinosas estrelladas en esta capa.

A la koniocortex, pertenecen las áreas de Brodmann 1-3 de la corteza sensorial somática, la zona 17 de la corteza visual y la zona 41 de la corteza auditiva. Todas estas áreas se comportan como mapas topográficos que cambian sus límites y campos receptivos de acuerdo con la experiencia sensorial.

La actividad en la koniocortex está regida por una serie de principios comunes de procesamiento de información. Uno de estos principios es la competencia, fue verificado experimentalmente que solamente unas pocas neuronas estrelladas de la koniocortex pertenecientes a la capa IV están activas cuando llega un estímulo. Este tipo de comportamiento se suele denominar "el ganador se lleva todo", (*winner-take-all* -WTA-), comportamiento en el que la neurona más activa "gana". A pesar de que en la gran mayoría de redes neuronales artificiales la neurona ganadora se identifica a través de un cálculo, en redes biológicas neuronales la neurona "ganadora" sale de un proceso dinámico basado en inhibiciones a las demás neuronas. La emergencia por tanto de una neurona depende tanto de inhibiciones laterales como de otra serie de procesos basados en las propiedades neuronales.

La verdadera novedad de este tipo de red se encuentra en la influencia en el comportamiento de ciertos mecanismos homeostáticos neuronales, entre ellos la metaplasticidad o la plasticidad intrínseca.

La plasticidad neuronal es la propiedad que emerge de la naturaleza y funcionamiento de las neuronas cuando estas establecen comunicación, y que modula la percepción de los estímulos del medio, tanto los que entran como los que salen. Esta dinámica deja una huella al tiempo que modifica la eficacia de la transferencia de la información a nivel de los elementos más finos del sistema. Dichas huellas son los elementos de construcción de la cosmovisión en donde lo anterior modifica la percepción de lo siguiente.

La KLN considera propiedades homeostáticas como metaplasticidad o la plasticidad intrínseca [9] que contribuyen a la dinámica de la WTA y hacer posible la competencia.

La ecuación que relaciona la entrada de la neurona  $O^j$  con su valor de salida  $O_j$  es la siguiente:

$$O_j = \frac{1}{1+e^{-k(net_j+0.5-2s^j)}} \quad (5)$$

Donde  $net_j$  es la red de salida de la neurona  $j$ .

La plasticidad intrínseca se calcula mediante la siguiente ecuación, que relaciona la función de activación  $s$  en el instante  $t$ :

$$s'_t = \frac{v \cdot O_{t-1} + s'_{t-1}}{v+1} \quad (6)$$

Donde  $v$  es pequeño factor arbitrario que ajusta el cambio de velocidad en la función de activación y  $O$  es la salida de la neurona.

Para el caso de las neuronas altamente activas,  $s_j$  aumenta, provocando que la función de activación se desplace hacia la derecha para que las salidas de las neuronas sean reguladas en el futuro. Para el caso de las neuronas menos activas, la función de activación se desplaza hacia la izquierda.

La curva de la plasticidad es la siguiente, para valores más altos de plasticidad la curva se alarga por la derecha.

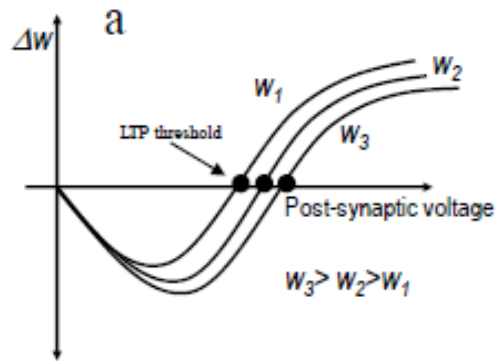


Figura 18. Curva de la plasticidad

La activación de las neuronas sigue la siguiente gráfica, dependiendo de la plasticidad intrínseca de las neuronas, la función de activación se desplazará hacia la izquierda o derecha. Si la entrada media de la neurona es baja, como en los casos A, B y C, la plasticidad intrínseca irá gradualmente desplazándose hacia la izquierda. Si la entrada es alta la curva de activación irá desplazándose hacia la derecha y dicha neurona no se saturará.

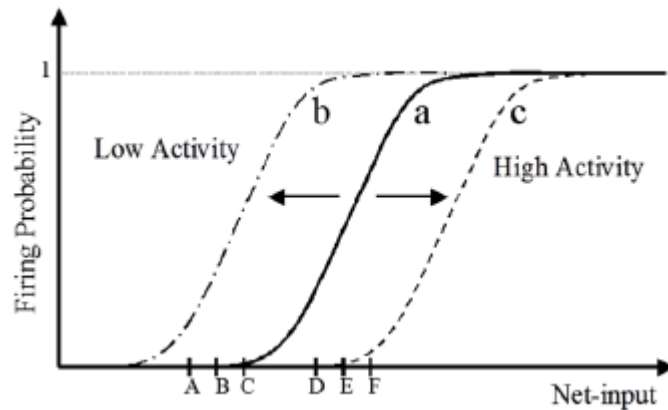


Figura 19. Curva de activación de las neuronas

La red KLN fue aplicada a la identificación de diez caracteres numéricos diferentes. Cada uno de ellos era representado en una matriz de pixeles de 5x3, donde los pixeles activos eran puestos a 1 mientras que los inactivos a 0. Las neuronas entradas son desde I1 hasta I15, cuyas salidas son los valores de los pixeles TC1...TC15, representando las neuronas tálamo-corticales. S1...S10 son las neuronas estrelladas. Las neuronas B1...B10 son neuronas inhibitorias que poseen plasticidad intrínseca y sinapsis no modificable.

Esta red, aplicada para el caso de los números funcionaba perfectamente, sin embargo, cuando se probó el conjunto de datos de Iris no fue capaz de organizarlos. Incluso modificando el factor de aprendizaje seguía sin dar resultados coherentes. Esta red necesita ser depurada para poder ser utilizada en un futuro.

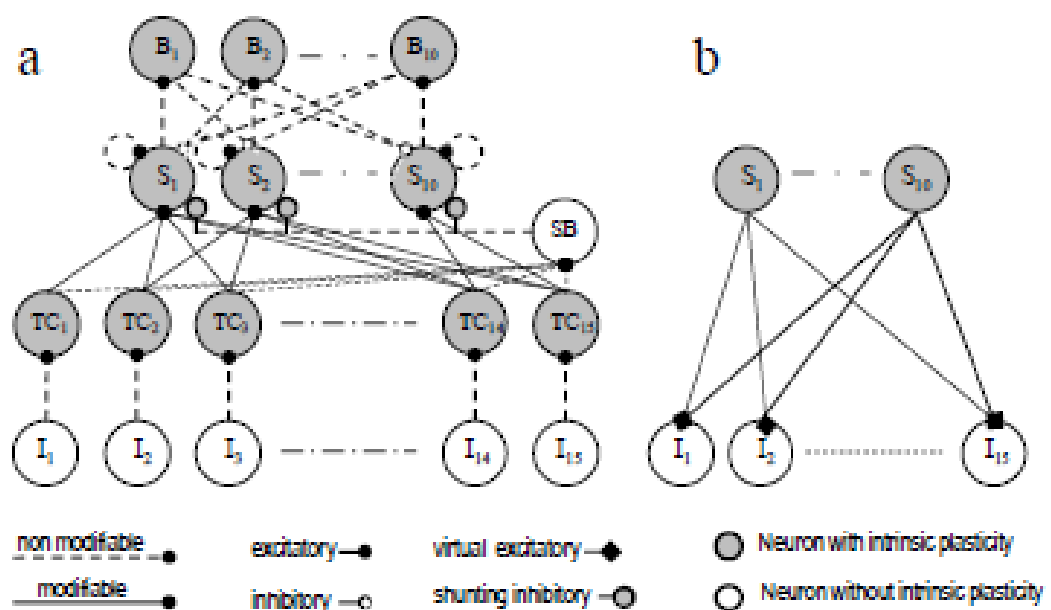


Figura 20. Red KLN

Este tipo de red evolucionó desde una red muy básica hasta ir introduciendo gradualmente nuevos elementos y sustituyendo los elementos con características no biológicas por unos con características de tipo biológico.

Al final de este proceso de diseño se ha llegado a la red KLN, que se asemeja a la cuarta capa de la *koniocortex*.

Las características principales de este tipo de red son la plasticidad intrínseca y la metaplasticidad, ambas características completan a la inhibición lateral para llevar a cabo el método “ganador se lleva todo” (WTA) así como procesos de aprendizaje.

## 7. Conclusión

Gracias a este proyecto he tratado diversas líneas de investigación así como de mi propio interés. La primera de ellas sería la red neuronal, partiendo de la base biológico con la composición de una neurona hasta el proceso sináptico.

La segunda sería como “traducir” una red neuronal biológica en una red neuronal artificial mediante un simple programa de ordenador. Partiendo de la estructura para ir incorporando nuevas propiedades biológicas para asemejar cada vez más esa red al propio cerebro.

Por otro lado, destacar la aplicación de este tipo de redes al análisis de datos bancarios para poder obtener información de cara a sacar mayor rentabilidad en el negocio. En mi caso he utilizado los datos de crédito de Alemania y Australia, obtenido una tasa de acierto de cerca del 72% y de fallo del 12%. Esto es un éxito, puesto que la red de Kohonen es muy simple y llevando a cabo una mejora de ella podría implementarse una red con una muy alta tasa de acierto.

Este tipo de redes aportan rapidez y facilidad para el manejo y categorización de los datos.

Este proyecto se podría resumir saber aplicar las redes neuronales artificiales para obtener mayor información a partir de una serie de datos bancarios, de modo fiable y a su vez rápido.

Claramente una línea de desarrollo de este proyecto sería lo explicado anteriormente relacionado con la tasa de fallo, puesto que no es lo mismo negarle el préstamo a un cliente “bueno” que concederle un préstamo a un cliente “malo”.

Como se puede ver, mediante este proyecto he conseguido adentrarme en el mundo financiero a través de la rama de *Business Intelligence*. Asimismo, también he tenido un acercamiento con el Big Data puesto que manejo cerca del

millar de datos, pero para tratarse de Big Data se extrapolarían los métodos a un conjunto de datos no uniforme y mucho mayor.

## 8. Referencias

1. Warren S. McCulloch and Walter Pitts: *A logical calculus of the ideas immanent in nervous activity*. University of Illinois, College of Medicine (1948)
2. H.D. Block, B.W. Knight, F. Rosenblatt: *Analysys of a four layer series-coupled perceptron*. Rev. Mod. Phys., 34 (1962)
3. Marvin Minsky and Seymour Papert: *Perceptrons: an introduction to computational geometry*. M.I.T. Press, Cambridge, Mass. (1969)
4. Bernard Widrow and Michael A. Lehr: *Perceptrons, Adalines and Backpropagation*. Stanford University (1960)
5. James Anderson and Edward Rosenfeld: *Neurocomputing*. The MIT Press (1989)
6. T.Kohonen and Panu Somervuo: *Neural Networks Research Centre*, P.O. Box 2200. FIN-02015 HUT, Finland (1998)
7. Gail A. Carpenter and Stephen Grossberg: *Adaptative Resonance Theory*. Springer, Boston (2011)
8. J. J. Hopfield: *Neural networks and physical systems with emergent collective computational abilities*. National Academy of Sciences of the USA (1982)
9. Francisco Javier Ropero Peláez y Diego Andina de la Fuente: *Intrinsic plasticity for natural competition in koniocortex-like neural networks*. Universidade Federal do ABC, Sao Paulo. Technical University of Madrid, ETSI Telecomunicación. (2016)
10. Howard Dresner: *The performance Management Revolution: Business Results through insight and action*. San Francisco (2008)

Otros:

Alexis Marcano-Cedeño, A. Marin-de-la-Barcena, J. Jiménez Trilo, J. A. Piñuela y D. Andina “Artificial Metaplasticity Neural Network Applied to CreditScoring.” (Junio 2016)

Análisis Business Intelligence “Varios artículos” (Mayo – Junio 2010)

<http:// analisisbi.blogspot.com/>

Emilio Arias *“Éxito en la implantación de un sistema Business Intelligence”*  
(Mayo 2010)